

N90-13417

SIMULTANEOUS QUATERNION ESTIMATION (QUEST) AND BIAS DETERMINATION

F. Landis Markley
Attitude Analysis Section,
Flight Dynamics Analysis Branch, Code 554.1
Goddard Space Flight Center,
Greenbelt, MD 20771

This paper presents tests of a new method for the simultaneous estimation of spacecraft attitude and sensor biases, based on a quaternion estimation algorithm minimizing Wahba's loss function. The new method is compared with a conventional batch least-squares differential correction algorithm. The estimates are based on data from strapdown gyros and star trackers, simulated with varying levels of Gaussian noise for both inertially-fixed and Earth-pointing reference attitudes. Both algorithms solve for the spacecraft attitude and the gyro drift rate biases. They converge to the same estimates at the same rate for inertially-fixed attitude, but the new algorithm converges more slowly than the differential correction for Earth-pointing attitude. The slower convergence of the new method for non-zero attitude rates is believed to be due to the use of an inadequate approximation for a partial derivative matrix. The new method requires about twice the computational effort of the differential correction. Improving the approximation for the partial derivative matrix in the new method is expected to improve its convergence at the cost of increased computational effort.

Introduction

When determining the three-axis attitude of a spacecraft, it is often necessary to simultaneously estimate sensor biases and misalignments. An extended Kalman filter or a batch least-squares differential correction procedure is generally used for this process [1]. These methods, collectively referred to as state estimation methods, require that the nonlinear estimation problem be linearized about *a priori* estimates of the attitude, biases, and misalignments. The purpose of this paper is to compare the standard batch least-squares differential correction procedure with a new algorithm [2] based on the *q*-method for minimizing Wahba's least-squares loss function [3]. The new algorithm computes the parameters iteratively, but does not linearize about an *a priori* attitude estimate, so it is expected to be more robust than the usual state estimation methods if the problem is highly nonlinear or if initial estimates are poor.

The development of the new algorithm is presented in detail in reference 2, along with some historical background, so it will not be repeated here. The following iterative procedure estimates the attitude at time t and the vector \mathbf{x} comprising the m non-attitude parameters. An *a priori* estimate \mathbf{x}^0 of \mathbf{x} is used to compute the matrix

$$B(t, \mathbf{x}) \equiv \sum_{i=1}^n a_i \Phi(t, t_i; \mathbf{x}) \mathbf{b}_i(\mathbf{x}) \mathbf{r}_i^T(\mathbf{x}), \quad (1)$$

where the unit vectors \mathbf{r}_i are representations in an inertial reference frame of the directions to some observed objects, the \mathbf{b}_i are the unit vector representations of the corresponding observations in the spacecraft body frame, the a_i are positive weights, and n is the number of observations. The 3×3 attitude propagation matrix $\Phi(t, t_0; \mathbf{x})$ is the solution of the differential equation

$$d\Phi(t, t_0; \mathbf{x})/dt = [\omega(t, \mathbf{x}) \times] \Phi(t, t_0; \mathbf{x}) \quad (2)$$

with initial value

$$\Phi(t_0, t_0; \mathbf{x}) = I \equiv \text{the } 3 \times 3 \text{ identity matrix}, \quad (3)$$

where the column vector $\omega(t, \mathbf{x})$ contains the components in the body frame of the spacecraft angular velocity relative to inertial space. The matrix $[\mathbf{v} \times]$ is defined for an arbitrary three-vector \mathbf{v} by

$$[\mathbf{v} \times] \equiv \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \quad (4)$$

The parameters in \mathbf{x} may enter the matrix $B(t, \mathbf{x})$ through the kinematics expressed by $\Phi(t, t_i; \mathbf{x})$, the observation modeling in $\mathbf{b}_i(\mathbf{x})$, or the reference vector models in $\mathbf{r}_i(\mathbf{x})$. The m matrices $\partial B / \partial x_j$, and the $m(m+1)/2$ independent matrices $\partial^2 B / \partial x_j \partial x_k$ expressing the derivatives of $B(t, \mathbf{x})$ with respect to the parameters must also be computed.

Standard methods [4] are next used to compute the largest eigenvalue $\lambda_{\max}(\mathbf{x})$ and corresponding normalized eigenvector $\mathbf{q}_{opt}(t, \mathbf{x})$ of the symmetric 4×4 matrix

$$K(t, \mathbf{x}) \equiv \begin{bmatrix} B(t, \mathbf{x}) + B^T(t, \mathbf{x}) - I \operatorname{tr} B(t, \mathbf{x}) & \mathbf{p}(t, \mathbf{x}) \\ \mathbf{p}^T(t, \mathbf{x}) & \operatorname{tr} B(t, \mathbf{x}) \end{bmatrix} \quad (5)$$

with the three-component column vector $\mathbf{p}(t, \mathbf{x})$ defined by

$$[\mathbf{p}(t, \mathbf{x}) \times] = B^T(t, \mathbf{x}) - B(t, \mathbf{x}). \quad (6)$$

Then the optimal attitude matrix for parameter vector \mathbf{x} , $A_{opt}(t, \mathbf{x})$, is computed from $\mathbf{q}_{opt}(t, \mathbf{x})$ by

$$A_{opt}(t, \mathbf{x}) = (q^2 - \mathbf{Q}^T \mathbf{Q}) I + 2\mathbf{Q} \mathbf{Q}^T - 2q[\mathbf{Q} \times], \quad (7)$$

where the three-vector part \mathbf{Q} and scalar part q of the quaternion $\mathbf{q}_{opt}(t, \mathbf{x})$ are given by

$$\mathbf{q}_{opt}^T(t, \mathbf{x}) = [\mathbf{Q}^T, q]. \quad (8)$$

The parameter vector update is given by

$$\delta \mathbf{x}_{opt} = W^{-1}(\mathbf{x})[\mathbf{h}(\mathbf{x}) - W^0(\mathbf{x} - \mathbf{x}^0)], \quad (9)$$

where, for $j, k = 1, \dots, m$,

$$W_{jk}(\mathbf{x}) = [W^0 - N^T(t, \mathbf{x})M^{-1}(t, \mathbf{x})N(t, \mathbf{x})]_{jk} - \text{tr}[A_{opt}^T(t, \mathbf{x})\partial^2 B(t, \mathbf{x})/\partial x_j \partial x_k], \quad (10)$$

$$h_j(\mathbf{x}) = \text{tr}\{[\partial B(t, \mathbf{x})/\partial x_j]A_{opt}^T(t, \mathbf{x})\}, \quad (11)$$

$$N_{1j}(t, \mathbf{x}) = \{[\partial B(t, \mathbf{x})/\partial x_j]A_{opt}^T(t, \mathbf{x})\}_{23} - \{[\partial B(t, \mathbf{x})/\partial x_j]A_{opt}^T(t, \mathbf{x})\}_{32}, \quad (12a)$$

$$N_{2j}(t, \mathbf{x}) = \{[\partial B(t, \mathbf{x})/\partial x_j]A_{opt}^T(t, \mathbf{x})\}_{31} - \{[\partial B(t, \mathbf{x})/\partial x_j]A_{opt}^T(t, \mathbf{x})\}_{13}, \quad (12b)$$

$$N_{3j}(t, \mathbf{x}) = \{[\partial B(t, \mathbf{x})/\partial x_j]A_{opt}^T(t, \mathbf{x})\}_{12} - \{[\partial B(t, \mathbf{x})/\partial x_j]A_{opt}^T(t, \mathbf{x})\}_{21}, \quad (12c)$$

and

$$M(t, \mathbf{x}) = \lambda_{max}(\mathbf{x}) I - B(t, \mathbf{x})A_{opt}^T(t, \mathbf{x}). \quad (13)$$

In these equations \mathbf{x}^0 is the *a priori* estimate of \mathbf{x} and W^0 is a symmetric positive-semidefinite matrix of weights assigned to this estimate; it is permissible to assign zero weights to the *a priori* estimate by setting $W^0 = 0$. The update $\delta \mathbf{x}_{opt}$ is added to \mathbf{x} to get the new parameter estimate. If the update is small enough, the procedure is complete; otherwise the computations are repeated from equation (1) until convergence is achieved.

The attitude covariance $P_{\theta\theta}$, the parameter covariance $P_{\mathbf{x}\mathbf{x}}$, and the cross-covariance $P_{\theta\mathbf{x}}$ of the converged estimate can be computed as follows:

$$P_{\theta\theta}(t) = \sigma_{tot}^2 [M^{-1}(t, \mathbf{x}) + M^{-1}(t, \mathbf{x})N(t, \mathbf{x})W^{-1}(\mathbf{x})N^T(t, \mathbf{x})M^{-1}(t, \mathbf{x})], \quad (14)$$

$$P_{\mathbf{x}\mathbf{x}} = \sigma_{tot}^2 W^{-1}(\mathbf{x}), \quad (15)$$

and

$$P_{\theta\mathbf{x}}(t) = P_{\mathbf{x}\theta}^T(t) = \sigma_{tot}^2 M^{-1}(t, \mathbf{x})N(t, \mathbf{x})W^{-1}(\mathbf{x}), \quad (16)$$

where

$$\sigma_{tot}^2 \equiv \left[\sum_{i=1}^n \sigma_i^{-2} \right]^{-1} \quad (17)$$

with σ_i^2 equal to the variance of the i^{th} vector measurement. The covariance computation assumes the weights to be

$$a_i = \sigma_{tot}^2 / \sigma_i^2 \quad (18)$$

and

$$W^0 = \sigma_{tot}^2 (P^0)^{-1}, \quad (19)$$

where P^0 is the covariance of the *a priori* parameter vector estimate. An expression for $M^{-1}(t, \mathbf{x})$ is derived in the Appendix, eliminating the need for a numerical matrix inversion.

Application to Gyro Drift Determination

For the application to be treated in this paper, we assume that the kinematic information for attitude propagation is obtained from three gyros aligned with the spacecraft body axes. In this case the estimation algorithm assumes that the body rate vector $\omega(t, \mathbf{x})$ is

$$\omega(t, \mathbf{x}) = \omega_g(t) - \mathbf{x}, \quad (20)$$

where $\omega_g(t)$ is the column vector of gyro outputs and \mathbf{x} , a three-component vector of gyro drifts, is the vector of parameters to be estimated. These parameters enter $B(t, \mathbf{x})$ through the attitude propagation matrices $\Phi(t, t_0; \mathbf{x})$. The first and second partial derivatives of $\Phi(t, t_0; \mathbf{x})$ with respect to the components of \mathbf{x} are needed to evaluate the corresponding partial derivatives of $B(t, \mathbf{x})$. The partial derivative of equation (2) with respect to x_j is, using equation (20),

$$d[\partial\Phi(t, t_0; \mathbf{x})/\partial x_j]/dt = -[\omega(t, \mathbf{x}) \times][\partial\Phi(t, t_0; \mathbf{x})/\partial x_j] + [\mathbf{e}_j \times] \Phi(t, t_0; \mathbf{x}), \quad (21)$$

where \mathbf{e}_j is the unit vector along the j^{th} spacecraft axis. The solution of this differential equation consistent with equations (2) and (3) is

$$\partial\Phi(t, t_0; \mathbf{x})/\partial x_j = \int_{t_0}^t \Phi(t, t'; \mathbf{x})[\mathbf{e}_j \times] \Phi(t', t_0; \mathbf{x}) dt'. \quad (22)$$

Using the group property and orthogonality of the attitude propagation matrix,

$$\Phi(t', t_0; \mathbf{x}) = \Phi(t', t; \mathbf{x})\Phi(t, t_0; \mathbf{x}) = \Phi^T(t, t'; \mathbf{x})\Phi(t, t_0; \mathbf{x}), \quad (23)$$

and the relation

$$\Phi[\mathbf{e}_j \times] \Phi^T = [(\Phi \mathbf{e}_j) \times], \quad (24)$$

which holds for any proper orthogonal matrix Φ , gives

$$\partial\Phi(t, t_0; \mathbf{x})/\partial x_j = -[\psi_j(t, t_0; \mathbf{x}) \times] \Phi(t, t_0; \mathbf{x}), \quad (25)$$

where $\psi_j(t, t_0; \mathbf{x})$ is the j^{th} column of the matrix

$$\Psi(t, t_0; \mathbf{x}) \equiv - \int_{t_0}^t \Phi(t, t'; \mathbf{x}) dt' = [\psi_1(t, t_0; \mathbf{x}) \quad \psi_2(t, t_0; \mathbf{x}) \quad \psi_3(t, t_0; \mathbf{x})]. \quad (26)$$

The matrices $\Phi(t, t_0; \mathbf{x})$ and $\Psi(t, t_0; \mathbf{x})$ are also used for the usual state estimation methods [5, 6]. They are computed by adding up contributions over time intervals $t_{i+1} - t_i$, which are chosen to be short enough that variations in ω over the interval can be neglected. Thus

$$\Phi(t_{i+1}, t_0; \mathbf{x}) = \Phi(t_{i+1}, t_i; \mathbf{x})\Phi(t_i, t_0; \mathbf{x}), \quad (27)$$

where

$$\Phi(t_{i+1}, t_i; \mathbf{x}) = I - [\mathbf{u} \times] \sin|\phi| + [\mathbf{u} \times]^2 (1 - \cos|\phi|), \quad (28)$$

\mathbf{u} is the unit vector

$$\mathbf{u} \equiv \omega/|\omega|, \quad (29)$$

and $|\phi|$ is the length of the vector

$$\phi \equiv \omega(t_{i+1} - t_i). \quad (30)$$

Similarly,

$$\Psi(t_{i+1}, t_0; \mathbf{x}) = \Psi(t_{i+1}, t_i; \mathbf{x}) + \Phi(t_{i+1}, t_i; \mathbf{x})\Psi(t_i, t_0; \mathbf{x}), \quad (31)$$

with

$$\Psi(t_{i+1}, t_i; \mathbf{x}) = -(t_{i+1} - t_i) \{ I - [\mathbf{u} \times] |\phi|^{-1} (1 - \cos|\phi|) + [\mathbf{u} \times]^2 (1 - |\phi|^{-1} \sin|\phi|) \}. \quad (32)$$

The second partial derivatives of $\Phi(t, t_0; \mathbf{x})$ only appear in $W(\mathbf{x})$ and not in $\mathbf{h}(\mathbf{x})$, so approximate forms will be used for these partials. They are also computed by adding up contributions over short time intervals $t_{i+1} - t_i$, giving

$$\begin{aligned} \partial^2 \Phi(t_{i+1}, t_0; \mathbf{x}) / \partial x_j \partial x_k &= [\partial^2 \Phi(t_{i+1}, t_i; \mathbf{x}) / \partial x_j \partial x_k] \Phi(t_i, t_0; \mathbf{x}) \\ &+ [\partial \Phi(t_{i+1}, t_i; \mathbf{x}) / \partial x_j] [\partial \Phi(t_i, t_0; \mathbf{x}) / \partial x_k] + [\partial \Phi(t_{i+1}, t_i; \mathbf{x}) / \partial x_k] [\partial \Phi(t_i, t_0; \mathbf{x}) / \partial x_j] \\ &+ \Phi(t_{i+1}, t_i; \mathbf{x}) [\partial^2 \Phi(t_i, t_0; \mathbf{x}) / \partial x_j \partial x_k], \end{aligned} \quad (33)$$

where the first partial derivatives are given by equations (25) – (32) and where

$$\begin{aligned} \partial^2 \Phi(t_{i+1}, t_i; \mathbf{x}) / \partial x_j \partial x_k &= (t_{i+1} - t_i)^2 \{ \frac{1}{2} (\mathbf{e}_j \mathbf{e}_k^T + \mathbf{e}_k \mathbf{e}_j^T - 2\delta_{jk} I) \\ &+ \frac{1}{3} (\phi_j [\mathbf{e}_k \times] + \phi_k [\mathbf{e}_j \times] + \delta_{jk} [\phi \times]) \}. \end{aligned} \quad (34)$$

The approximation is in equation (34), which is valid to first order in ϕ . Starting the iterative computations of equations (27), (31), and (33) requires initial values for the matrices: the identity for $\Phi(t_0, t_0; \mathbf{x})$ from equation (3), and zero for $\Psi(t_0, t_0; \mathbf{x})$ and $\partial^2 \Phi(t_0, t_0; \mathbf{x}) / \partial x_j \partial x_k$.

Observation Modeling

Star tracker data are used to estimate the spacecraft attitude and gyro drifts. Each star tracker measurement is a two-component vector \mathbf{y}_i giving the location of the star image in the focal plane of the sensor. For attitude estimation with the new method we need to compute the star unit vector in the spacecraft body frame \mathbf{b}_i in terms of the measurement data \mathbf{y}_i . The star unit vector in the sensor frame is

$$\mathbf{s}_i = (1 + |\mathbf{y}_i|^2)^{-1/2} [(\mathbf{y}_i)_1, (\mathbf{y}_i)_2, 1]^T, \quad (35)$$

and then \mathbf{b}_i is given by

$$\mathbf{b}_i = C_i^T \mathbf{s}_i, \quad (36)$$

where C_i is the proper orthogonal 3×3 matrix defining the orientation in the body frame of the star tracker making this observation.

Data Simulation

Simulated gyro data and star tracker data are used to test the algorithm. The simulation assumes a constant angular velocity vector ω_{true} . The gyro data are simulated by adding varying levels of Gaussian noise to the components of ω_{true} . A true attitude matrix is computed by integrating the angular rates;

$$dA_{true}(t)/dt = -[\omega_{true} \times] A_{true}(t), \quad (37)$$

with some specified initial attitude matrix $A_{true}(t_0)$.

A star is initially simulated for each star tracker by randomly generating a measurement vector y_i within the star tracker field of view. Equations (35) and (36) then give the star unit vector \mathbf{b}_i in the body frame, and the star unit vector in the inertial reference frame is given by

$$\mathbf{r}_i = A_{true}^T(t_i) \mathbf{b}_i, \quad (38)$$

where t_i is the simulation time. For successive simulation times t_i , the reference vector \mathbf{r}_i is held fixed and the vector in the body frame is computed as

$$\mathbf{b}_i = A_{true}(t_i) \mathbf{r}_i. \quad (39)$$

Then the corresponding vector in the star tracker reference frame is given by the inverse of equation (36),

$$\mathbf{s}_i = C_i \mathbf{b}_i; \quad (40)$$

the measurement vector y_i by the inverse of equation (35),

$$y_i = (s_i)_3^{-1} [(s_i)_1, (s_i)_2]^T, \quad (41)$$

and Gaussian noise is added to the two components of y_i . This process is continued until the star has been tracked for some fixed number of observations or until it leaves the field of view, at which time a new star is randomly placed in the field of view. Earth and Sun interference are neglected in these simulations.

Comparison Algorithm

The algorithm chosen for comparison is a batch least-squares differential correction algorithm similar to that employed in the attitude ground support system of the Upper Atmosphere Research Satellite (UARS) [6]. The algorithm provides a least-squares estimate of a six-component state vector

$$\delta \mathbf{X}^T \equiv [\delta \theta^T \delta \mathbf{x}^T], \quad (42)$$

where $\delta \theta$ is the attitude error vector at epoch, and $\delta \mathbf{x}$ is the error in the gyro drift estimates. This is updated iteratively as follows. At the start of each iteration, an estimate \mathbf{x} of the gyro drifts and $A_{est}(t_0)$ of the epoch attitude are available. For each measurement y_i , a predicted value \mathbf{g}_i is computed from the known reference vector \mathbf{r}_i by equations identical to (39) – (41), but with the unknown attitude matrix $A_{true}(t_i)$ replaced by

$$A_{est}(t_i) = \Phi(t_i, t_0; \mathbf{x}) A_{est}(t_0). \quad (43)$$

The computed value \mathbf{g}_i is seen to depend on both \mathbf{x} and $A_{est}(t_0)$. The optimal state update is the solution of

$$F \delta \mathbf{X} = \sum_{i=1}^n \sigma_i^{-2} G_i^T (y_i - \mathbf{g}_i) - \begin{bmatrix} \mathbf{0} \\ (P^0)^{-1} (\mathbf{x} - \mathbf{x}^0) \end{bmatrix}, \quad (44)$$

where

$$F \equiv \sum_{i=1}^n \sigma_i^{-2} G_i^T G_i + \begin{bmatrix} 0 & 0 \\ 0 & (P^0)^{-1} \end{bmatrix}, \quad (45)$$

with σ_i^{-2} , P^0 , and \mathbf{x}^0 as defined previously, and with $\mathbf{0}$ a three-vector of zeros and $\mathbf{0}$ a 3×3 matrix of zeros. The 2×6 matrix G_i of partial derivatives of the errors of the i^{th} measurement with respect to $\delta \mathbf{X}$ is given by

$$G_i = \begin{bmatrix} (y_i)_1(y_i)_2 & -1 - (y_i)_1^2 & (y_i)_2 \\ 1 + (y_i)_2^2 & -(y_i)_1(y_i)_2 & -(y_i)_1 \end{bmatrix} C_i [\Phi(t_i, t_0; \mathbf{x}) \quad \Psi(t_i, t_0; \mathbf{x})]. \quad (46)$$

This state update gives new estimates of the gyro drifts and attitude:

$$\mathbf{x}_{new} = \mathbf{x} + \delta \mathbf{x}, \quad (47)$$

and

$$A_{new}(t_0) = \{I - |\delta \theta|^{-1}[\delta \theta \times] \sin|\delta \theta| + |\delta \theta|^{-2}[\delta \theta \times]^2(1 - \cos|\delta \theta|)\} A_{est}(t_0). \quad (48)$$

This iterative procedure is repeated until convergence is achieved. An estimate of the covariance matrix is provided by

$$\begin{bmatrix} P_{\theta\theta} & P_{\theta\mathbf{x}} \\ P_{\mathbf{x}\theta} & P_{\mathbf{x}\mathbf{x}} \end{bmatrix} = F^{-1}. \quad (49)$$

The initial attitude to begin the first iteration is provided by the q -method, as embodied in equations (1) - (7).

Numerical Examples

Tests were performed for both inertially-fixed and earth-pointing spacecraft attitudes, with star tracker orientations and other parameters corresponding to the Gamma Ray Observatory (GRO) [7] and UARS [6] spacecraft, respectively. Two star trackers were modeled with 8 degree by 8 degree fields of view and with an angle of approximately 73 degrees between their boresights. Some tests were performed with perfect star tracker measurements, but the results presented in this paper all include Gaussian noise on each star tracker output with standard deviation of 8 arc seconds, or 3.88×10^{-5} radians. The time interval between star tracker measurements was 32.768 seconds, the interval used by the UARS onboard computer. The data were simulated with no gyro bias, and the estimations were performed with non-zero initial bias estimates, so the bias estimate is the same as the bias error for these tests. The initial bias error was 10^{-4} radians/sec along either the spacecraft roll or yaw axis, but only representative results with initial roll bias errors are given below.

The epoch time t_0 for the estimation was taken to be the time of the first observation. In all but four tests, the true attitude matrix at epoch was set equal to the identity matrix. The tests for one simulation case were repeated with four different true attitude matrices at epoch:

$$A_{true}(t_0) = \begin{bmatrix} 0.352 & 0.864 & 0.360 \\ -0.864 & 0.152 & 0.480 \\ 0.360 & -0.480 & 0.800 \end{bmatrix}, \quad (50a)$$

$$A_{true}(t_0) = \text{diag}[1, -1, -1], \quad (50b)$$

$$A_{true}(t_0) = \text{diag}[-1, 1, -1], \quad (50c)$$

and

$$A_{true}(t_0) = \text{diag}[-1, -1, 1], \quad (50d)$$

where $\text{diag}[\dots]$ denotes a 3×3 matrix with the given elements on the main diagonal and zeros elsewhere. The attitude and bias errors for these different initial attitudes were identical to those for $A_{true}(t_0) = I$ within the precision of the output, as they should be. The covariance P^0 of the *a priori* bias estimates was taken to be infinite for all tests.

A representative subset of the tests is presented in Tables 1-7. At least 10 iterations were performed in each case, and the errors for all the iterations after those presented are identical to the errors of the last iteration in the table, to the precision of the table. The first "iteration" in the differential correction (DC) columns is not really a DC iteration; it is an initial attitude estimation using the *q*-method, as explained above. Thus the bias error after one DC "iteration" is the *a priori* error. The last line in each table is the estimate of the error standard deviations from the covariance matrices of equations (14) and (15) or equation (49).

Tables 1-6 present the tests with inertially-fixed attitude. These are in pairs: Tables 1 and 2 give the results for the highest observability case with two star trackers and a full orbit of data, Tables 3 and 4 have two star trackers but only 10 observations in each, and Tables 5 and 6 are for the case of only 10 observations in a single star tracker. Each simulated star was observed five times in these tests, so the cases in Tables 5 and 6 contain only two stars; the angular separation between these stars was 1.3 degrees. In each pair of tables, the first (odd-numbered) presents the results with no gyro noise, and the second (even-numbered) shows the effects of Gaussian noise on each gyro with standard deviation of 1 degree/hour, or 4.848×10^{-6} rad/sec.

The most important aspect of the tests, as concerns this paper, is the comparison of the results of the new method to those of the DC. The bias and attitude errors are not the same at each iteration, but both the general rate of convergence and the final converged errors are almost identical. Where there are differences, the errors of the new method are slightly lower, but not by a significant amount.

In the cases without gyro noise, the covariance matrix is a good indicator of the estimation errors. This correspondence is especially striking in Tables 1 and 3, while the actual errors in Table 5 are about 20 times less than the covariance matrix would indicate. The errors in the latter case are remarkably small considering the poor measurement geometry, with only two reference vectors separated by 1.3 degrees. When gyro noise is included, the actual errors can exceed the covariance estimates; this is not surprising since the covariance computation does not take gyro errors into account, nor does any other part of the estimation process. When unrealistically large gyro noise with standard deviation of 100 degree/hour was included, both estimation procedures became unreliable. The new method failed catastrophically when the nominally positive-definite matrix W defined by equation (10) developed a negative element on its main diagonal. The DC did not become singular, since the matrix F of equation (45), unlike W , is manifestly positive-semidefinite; but the bias estimation error increased monotonically for the first 10 iterations. Thus the new method is somewhat less robust than the DC in this case; but this is not very significant since a Kalman filter or smoother should probably be used in the presence of large amounts of dynamic noise [5].

Table 1. Bias and Attitude Errors for Inertially-fixed Attitude
with Two Star Trackers, 95.6 Minutes of Data, and no Gyro Noise

Iteration	Batch DC		New Method	
	Bias (rad/sec)	Attitude (rad)	Bias (rad/sec)	Attitude (rad)
1	1.00D-4	2.88D-1	1.68D-6	2.88D-1
2	5.65D-7	1.50D-3	1.69D-9	4.83D-3
3	1.68D-9	5.83D-6	1.68D-9	5.82D-6
4	1.68D-9	5.81D-6	1.68D-9	5.80D-6
Covariance	2.90D-9	9.53D-6	2.91D-9	9.54D-6

Table 2. Bias and Attitude Errors for Inertially-fixed Attitude
with Two Star Trackers, 95.6 Minutes of Data, and Gyro Noise of 1 deg/hour

Iteration	Batch DC		New Method	
	Bias (rad/sec)	Attitude (rad)	Bias (rad/sec)	Attitude (rad)
1	1.00D-4	2.87D-1	1.73D-6	2.87D-1
2	6.54D-7	2.66D-3	2.90D-7	5.46D-3
3	2.90D-7	1.98D-3	2.90D-7	1.98D-3
Covariance	2.90D-9	9.53D-6	2.91D-9	9.54D-6

Table 3. Bias and Attitude Errors for Inertially-fixed Attitude
with Two Star Trackers, 5.5 Minutes of Data, and no Gyro Noise

Iteration	Batch DC		New Method	
	Bias (rad/sec)	Attitude (rad)	Bias (rad/sec)	Attitude (rad)
1	1.00D-4	1.48D-2	2.12D-7	1.48D-2
2	2.17D-7	2.95D-5	2.07D-7	3.02D-5
3	2.07D-7	2.93D-5	2.07D-7	2.93D-5
Covariance	2.10D-7	3.67D-5	2.10D-7	3.68D-5

Table 4. Bias and Attitude Errors for Inertially-fixed Attitude
with Two Star Trackers, 5.5 Minutes of Data, and Gyro Noise of 1 deg/hour

Iteration	Batch DC		New Method	
	Bias (rad/sec)	Attitude (rad)	Bias (rad/sec)	Attitude (rad)
1	1.00D-4	1.49D-2	3.50D-6	1.49D-2
2	3.50D-6	2.71D-4	3.50D-6	2.71D-4
3	3.50D-6	2.72D-4	3.50D-6	2.72D-4
Covariance	2.10D-7	3.67D-5	2.10D-7	3.68D-5

Table 5. Bias and Attitude Errors for Inertially-fixed Attitude
with One Star Tracker, 5.5 Minutes of Data, and no Gyro Noise

	Batch DC		New Method	
Iteration	Bias (rad/sec)	Attitude (rad)	Bias (rad/sec)	Attitude (rad)
1	1.00D-4	1.63D-2	1.14D-4	1.63D-2
2	2.29D-5	3.25D-3	1.07D-6	5.69D-3
3	1.38D-6	2.94D-4	1.08D-6	2.49D-4
4	1.10D-6	2.55D-4	1.08D-6	2.51D-4
5	1.11D-6	2.55D-4	1.08D-6	2.51D-4
Covariance	2.39D-5	4.18D-3	2.40D-5	4.19D-3

Table 6. Bias and Attitude Errors for Inertially-fixed Attitude
with One Star Tracker, 5.5 Minutes of Data, and Gyro Noise of 1 deg/hour

	Batch DC		New Method	
Iteration	Bias (rad/sec)	Attitude (rad)	Bias (rad/sec)	Attitude (rad)
1	1.00D-4	1.44D-2	1.18D-4	1.44D-2
2	2.25D-5	1.16D-2	4.39D-6	7.11D-4
3	4.51D-6	1.54D-2	4.41D-6	1.54D-2
4	4.50D-6	1.54D-2	4.41D-6	1.54D-2
Covariance	2.37D-5	4.16D-3	2.38D-5	4.18D-3

Table 7. Bias and Attitude Errors for Earth-Pointing Attitude
with One Star Tracker, 5.5 Minutes of Data, and no Gyro Noise

	Batch DC		New Method	
Iteration	Bias (rad/sec)	Attitude (rad)	Bias (rad/sec)	Attitude (rad)
1	1.00D-4	3.01D-2	8.04D-5	3.01D-2
2	1.37D-5	2.54D-3	1.73D-5	3.02D-2
3	4.74D-6	8.52D-4	7.17D-6	3.05D-3
4	4.88D-6	8.76D-4	5.31D-6	1.27D-3
5	4.88D-6	8.76D-4	4.97D-6	9.48D-4
6	4.88D-6	8.76D-4	4.91D-6	8.90D-4
7	4.88D-6	8.76D-4	4.89D-6	8.79D-4
8	4.88D-6	8.76D-4	4.89D-6	8.78D-4
9	4.88D-6	8.76D-4	4.89D-6	8.77D-4
Covariance	1.24D-5	2.15D-3	1.12D-5	1.96D-3

The attitude errors in Table 2 are larger than those in Table 4, showing that estimators that do not handle dynamic noise correctly should not be used with long spans of data including the effects of dynamic noise.

The tests with Earth-pointing attitude used a constant pitch rate of -1.083073×10^{-3} rad/sec, which corresponds to an orbit period of 96.7 minutes. For these tests, a simulated star was tracked until it left the star tracker field of view. The test parameters were otherwise the same as for inertially-fixed attitude. The new method did not fare as well in these tests; it generally required more iterations than the DC to converge, although the final errors were virtually identical. This suggests the presence of errors in the matrix $W(\mathbf{x})$ that steers the estimates to their optimal values, and not in the vector $\mathbf{h}(\mathbf{x})$ that identifies the optimum once it has been reached. Table 7 presents the results of a test with a single star tracker, observing only two stars with angular separation of 7 degrees. This is a particularly discouraging example, in which the DC converged in four iterations, while the new method required nine. Since this is a low observability case, in which attitude kinematics information is more important compared to the measurements than in a high observability case, an accurate computation of $W(\mathbf{x})$ is especially important.

The greater success of the new method for inertially-fixed attitude than for non-inertial attitude suggests the inadequacy of the approximation of equation (34) for the matrix of second partial derivatives $\partial^2 \Phi(t_{i+1}, t_i; \mathbf{x}) / \partial x_j \partial x_k$, which appears in $W(\mathbf{x})$ and not in $\mathbf{h}(\mathbf{x})$. This approximation should be replaced by one that is valid for all values of the rotation angle ϕ , subject to the assumption that the angular rates are approximately constant between observations. This may also avoid the failure of the new method in the test with 100 degree/hour gyro noise, since this has an effect on the propagation of the partial derivative matrices similar to the effects of actual angular rates of the same size.

The computational effort required by the two algorithms was also measured. Both algorithms were implemented in double-precision Fortran and executed on a DEC VAX 11/780. The CPU times were proportional to the number of iterations performed, the times per iteration for the two methods being

$$t_{CPU, new} = 13 + 15.2 n \text{ msec} \quad (51a)$$

and

$$t_{CPU, DC} = 31 + 6.7 n \text{ msec}. \quad (51b)$$

where n is the number of observations. The coefficient of n in these times can be interpreted as the time required to process a measurement, including propagation of the attitude transition matrix, partial derivative matrices, and so forth. The n -independent term represents the end-of-iteration computations, including matrix inversions and computation of updates to the bias vector and attitude matrix. Thus equation (51) shows that the measurement processing is more expensive for the new method, while the end-of-iteration computations of the DC require more effort. The exact CPU times will vary from case to case, but the DC appears to be about twice as fast, overall, as the new method for the numbers of measurements typically processed. Improving the computation of the matrix of second partial derivatives for the new method will require even more effort to process each measurement.

Conclusions

These tests establish the validity of a new method for the simultaneous estimation of spacecraft attitude and sensor biases, based on a quaternion estimation algorithm minimizing Wahba's loss function. The new algorithm performs as well as a batch least-squares differential correction in tests with inertially-fixed attitude, in the sense of converging to equally accurate estimates in the same number of iterations. The new algorithm converges more slowly than the differential correction for Earth-pointing attitude, probably owing to the use of an inadequate approximation for a partial derivative matrix in the new method. The new method does not show any advantages in terms of robustness or speed of convergence, and in addition requires about twice the computational effort of the differential correction. It is hoped that improving the approximation for the partial derivative matrix in the new method will improve its convergence and/or robustness, without adding significantly to its computational burden.

Appendix

The matrix $B(t, \mathbf{x})$ has the singular value decomposition [8]

$$B = U_+ S' V_+^T, \quad (\text{A1})$$

where U_+ and V_+ are proper orthogonal matrices, and

$$S' = \text{diag}[S_1, S_2, S_3], \quad (\text{A2})$$

a 3×3 matrix with S_1, S_2 , and S_3 on the main diagonal and zeros elsewhere. The arguments t and \mathbf{x} are omitted from this and all subsequent equations for notational simplicity. The optimal attitude estimate is given in terms of these matrices by [8]

$$A_{opt} = U_+ V_+^T. \quad (\text{A3})$$

The maximum eigenvalue λ_{max} of the matrix K defined by equation (5) is related to the optimal attitude by [2]

$$\lambda_{max} = \text{tr}(A_{opt} B^T), \quad (\text{A4})$$

where tr denotes the trace. Equations (A1) – (A4) and (13) show that

$$\det B = S_1 S_2 S_3, \quad (\text{A5})$$

$$\lambda_{max} = S_1 + S_2 + S_3, \quad (\text{A6})$$

$$M = U_+ \text{diag}[S_2 + S_3, S_3 + S_1, S_1 + S_2] U_+^T. \quad (\text{A7})$$

and

$$\det M = (S_2 + S_3)(S_3 + S_1)(S_1 + S_2). \quad (\text{A8})$$

We now define the scalar

$$\kappa \equiv \frac{1}{2} [\lambda_{max}^2 - \text{tr}(B B^T)]. \quad (\text{A9})$$

A little algebra shows that

$$\kappa = S_2 S_3 + S_3 S_1 + S_1 S_2, \quad (\text{A10})$$

$$\kappa \lambda_{\max} - \det B = \det M, \quad (\text{A11})$$

and

$$\kappa I + BB^T = U_+ \text{diag}[(S_3 + S_1)(S_1 + S_2), (S_1 + S_2)(S_2 + S_3), (S_2 + S_3)(S_3 + S_1)]U_+^T = \text{adj } M, \quad (\text{A12})$$

where adj denotes the adjoint matrix. Equations (A11) and (A12) give the desired result

$$M^{-1} = (\kappa \lambda_{\max} - \det B)^{-1}(\kappa I + BB^T). \quad (\text{A13})$$

The evaluation of M^{-1} by means of equations (A9) and (A13) does not require the singular value decomposition of B .

References

- [1] WERTZ, JAMES R. (editor) *Spacecraft Attitude Determination and Control*, D. Reidel, Dordrecht, the Netherlands, 1978
- [2] MARKLEY, F. LANDIS "Attitude Determination and Parameter Estimation using Vector Observations: Theory," *Journal of the Astronautical Sciences*, Vol. 37, No. 1, January-March, 1989
- [3] WAHBA, GRACE "A Least Squares Estimate of Spacecraft Attitude," *SIAM Review*, Vol. 7, No. 3, July 1965, p. 409.
- [4] SHUSTER, M. D. and OH, S. D. "Three-Axis Attitude Determination from Vector Observations," *Journal of Guidance and Control*, Vol. 4, No. 1, January-February 1981, pp. 70-77
- [5] LEFFERTS, E. J., MARKLEY, F. L., and SHUSTER, M. D. "Kalman Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 5, No. 5, September-October 1982, pp. 417-429
- [6] KRACK, K. et al., *Upper Atmosphere Research Satellite (UARS) / Extreme Ultraviolet Explorer (EUV) Flight Dynamics Support System (FDSS) Functional Specifications*, Computer Sciences Corporation, CSC/TR-88/6032, November 1988
- [7] BLEJER, D. et al., *Gamma Ray Observatory (GRO) Flight Dynamics Support System (FDSS) Specifications: II Functional Specifications, Update 1*, Computer Sciences Corporation CSC/TR-85/6005UD1, February 1987
- [8] MARKLEY, F. LANDIS "Attitude Determination Using Vector Observations and the Singular Value Decomposition," *Journal of the Astronautical Sciences*, Vol. 36, No.3, July-September 1988, pp. 245-258

